Testing the Assumptions of Sequential Bifurcation for Factor Screening (revision of CentER DP 2015-034)

Shi, Wen; Kleijnen, J.P.C.

Publication date:
2017

Document Version
Early version, also known as pre-print

No. 2017-006

TESTING THE ASSUMPTIONS OF SEQUENTIAL BIFURCATION FOR FACTOR SCREENING

By

Wen Shi, Jack P.C. Kleijnen

30 January 2017

This is a revised version of
CentER Discussion Paper No. 2015-034

ISSN 0924-7815
ISSN 2213-9532

Testing the Assumptions of Sequential Bifurcation for Factor Screening

Wen Shi
School of Logistics and Engineering Management, Hubei University of Economics, Wuhan 430205, China, shi3wen@163.com

Jack P.C. Kleijnen
Department of Management/Center for Economic Research (CentER), Tilburg University, 5000 LE Tilburg, The Netherlands, kleijnen@uvt.nl

January 30, 2017

Sequential bifurcation (or SB) is an efficient and effective factor-screening method; i.e., SB quickly identifies the important factors (inputs) in experiments with simulation models that have very many factors—provided the SB assumptions are valid. The specific SB assumptions are: (i) a second-order polynomial is an adequate approximation (a valid metamodel) of the implicit input/output function of the underlying simulation model; (ii) the directions (signs) of the first-order effects are known (so the first-order polynomial approximation is monotonic); (iii) so-called "heredity" applies; i.e., if an input has no important first-order effect, then this input has no important second-order effects. Moreover—like many other statistical methods—SB assumes Gaussian simulation outputs if the simulation model is stochastic (random). A generalization of SB called "multiresponse SB" (or MSB) uses the same assumptions, but allows for simulation models with multiple types of responses (outputs). To test whether these assumptions hold, we develop new methods. We evaluate these methods through Monte Carlo experiments and a case study.

Key words: sensitivity analysis; experimental design; meta-modelling; validation; regression; simulation

JEL: C0, C1, C9, C15, C44

---

1. Introduction

By definition, *factor screening*—or briefly *screening*—means searching for the really important factors—or inputs—among the many factors that can be varied in an experiment with a given simulation model (we shall define "important" below). For example, Bettonvil and Kleijnen (1997) applies the screening method called "sequential bifurcation"—abbreviated to SB—to a case study, and finds that only 15 of the 281 inputs are really important. So, screening assumes that input effects are *sparse*; i.e., only a few inputs among the many inputs are really important. Related to

sparsity are the *Pareto* principle or the *20-80* rule, which implies that roughly 20% of the inputs account for 80% of the effect on the output. The *law of parsimony* or *Occam's razor* implies that a simpler explanation with fewer "factors" is preferred to a more complex explanation—all other things being equal. Altogether, we conclude that there is really a need for screening in the design and analysis of experiments with practical simulation models.

Furthermore, we assume that the number of inputs (say) k is so large that *classic designs*—such as resolution-III (R-III) designs—cannot be applied. For example, a R-III design requires an experiment with at least $k + 1$ input combinations to estimate the effects in a first-order polynomial with k inputs plus an intercept, assuming this polynomial provides an adequate (valid) approximation or *metamodel* of the simulation model. Higher-order polynomials require bigger designs; e.g., a second-order polynomial may be estimated through a central composite design (CCD), which has $1 + k + k(k-1)/2 + k$ input combinations. Kleijnen (2015) discusses *design of experiments* (DOE), including classic designs—such as R-III designs and CCDs—and several types of screening designs—besides SB. A recent publication that discusses screening designs is Shi et al. (2016).

In this article we focus on SB and the extension of SB to simulation models with multiple responses. The latter is called *multiresponse SB* (MSB) in Shi et al. (2014a), and includes SB as a special case; namely, a single response. For brevity's sake we shall write "MSB" instead of "SB or MSB" or "MSB including SB" if the context makes confusion unlikely. The goal of MSB is to identify the inputs that have important effects on one or more output (response) types among the $n \geq 1$ output types.

We consider the following problem. The given simulation model has so many inputs that application of classic designs would require too much computer time. Therefore the users of this simulation model decide to apply screening. Each type of screening design has its own assumptions. Because MSB is the most efficient screening design, the users decide to apply MSB. In general we emphasize that after users have applied a statistical method to solve a given problem, these users should next examine the results to verify whether these results are not conflicting with the assumptions of the method. For example, the users apply linear regression to analyze a data set obtained through a simulation experiment; then the users should next validate the estimated (fitted) regression model through the coefficient of determination R^2 and cross-validation (R^2 and cross-validation are detailed in Kleijnen (2015, p. 112-121). More specifically, *after* the users have applied MSB to find important inputs, these users should verify whether these results do not conflict with the assumptions of MSB. For this verification we derive and evaluate several statistical tests

in this article. These *post-screening* (follow-up) tests definitely require less experimentation than MSB requires; e.g., one test requires only two (extreme) combinations to validate the first-order polynomial metamodel in SB (which assumes a single response type).

SB was originally developed in Bettonvil (1990)'s dissertation and summarized in Bettonvil and Kleijnen (1997). Several authors extended SB; see the many references in Kleijnen (2015), and also see Han et al. (2017) and Martín and Sánchez (2015). To save space, we refer to the detailed description of SB and MSB in (Shi et al., 2014a); for our article it suffices to detail the following *three specific MSB assumptions*:

1. *Second-order polynomials* provide valid metamodels:

$$y^{(l)} = \beta_0^{(l)} + \sum_{j=1}^{k} \beta_j^{(l)} x_j + \sum_{j=1}^{k-1} \sum_{j'=j+1}^{k} \beta_{j;j'}^{(l)} x_j x_{j'} + \sum_{j=1}^{k} \beta_{j;j}^{(l)} x_j^2 + e^{(l)} \quad (1)$$

where $y^{(l)}$ denotes the metamodel's predictor for simulation output l with $l = 1, ..., n$ and $n \geq 1$ ($n = 1$ in SB), x_j the standardized (coded, scaled) simulation input j ($j = 1, \ldots, k$) so $-1 \leq x_j \leq 1$—if an original input is qualitative, then its levels are randomly associated with the standardized values -1 and 1—$\beta_0^{(l)}$ the intercept for output l, $\beta_j^{(l)}$ the first-order (or main effect) of x_j for output l, $\beta_{j;j'}^{(l)}$ the interaction between x_j and $x_{j'}$ for output l, $\beta_{j;j}^{(l)}$ the purely quadratic effect of x_j for output l, and $e^{(l)}$ the approximation error with zero mean for output l.

2. The $\beta_j^{(l)}$ have *known signs*, so that the low bound l_j and the upper bound u_j of the original (nonstandardized) input z_j can be defined such that all k first-order effects are nonnegative for one of the n output types—(say) output type 1 (the symbol l_j is an easy mnemonic for "low", but should not be confused with the symbol l in the superscript (l); we use the symbol (l) because Shi et al. (2014a) uses that symbol). This assumption implies $\beta_j^{(1)} \geq 0$ (the superscript is (1), not (l)). Without assumption 2, first-order effects may cancel each other within a group of individual inputs that is used in MSB (these input groups are detailed in Shi et al. (2014a)).

3. If input j has no first-order effect on simulation output l (so $\beta_j^{(l)} = 0$), then this input has no second-order effects on this output (so $\beta_{j;j}^{(l)} = 0$ and $\beta_{j;j'}^{(l)} = 0$ with $j' \neq j$). Wu and Hamada (2009) calls this the *heredity* assumption.

Many publications on screening discuss the plausibility of these three assumptions; we discuss formalized statistical tests.

Note: If our tests reject these MSB assumptions, then MSB may still identify the important inputs; i.e., these assumptions are sufficient but not necessary. However, we consider it to be unlikely

that these assumptions do not hold, but MSB still "works". Hasty readers may skip paragraphs that start with "Note:", and still understand this article.

Besides the preceding three specific MSB assumptions, MSB—like many other statistical methods—assumes that the simulation outputs have *normal* (Gaussian) distributions. Our tests also assume normality.

We organize the rest of this article as follows. In Section 2 we discuss the assumed normality of the simulation outputs. In Section 3 we detail our tests for the specific three MSB assumptions; in Section 3.4 we compare these tests through a Monte Carlo experiemnt that does satisfy all MSB assumptions. In Section 4 we compare these tests through a case study concerning a logistics system in China, which may not satisfy all the assumptions. In Section 5 we summarize the major conclusions and sketch future research.

2. Normality of simulation outputs

MSB assumes that the simulation outputs have normal distributions. The number of replications (say) m for a given input combination may be either constant or random. For a constant m, MSB uses a t_{m-1} statistic to test whether the sum of the first-order effects of a given input group is significantly higher than a threshold Δ; see Kleijnen (2015, pp. 149-150). It is well known that t_{m-1} is quite insensitive to nonnormality; i.e., t_{m-1} is quite "robust". Instead of using a constant m, MSB may apply Wan et al. (2010)'s *sequential probability ratio test* (SPRT) to determine a "good" m. This SPRT assumes normality. In general, SPRTs may also be robust, as detailed in Kleijnen and Shi (2017). Our tests of the three MSB assumptions also assume normal simulation outputs, and use t-statistics—as we shall see. For completeness' sake we show how the users can easily test normality in MSB, as follows.

The case study involving a logistic system in China illustrates that Wan et al. (2010)'s SPRT requires higher values for m as the search continues so groups of individual inputs are split into subgroups (these subgropups have smaller signal / noise ratios). Actually, this case study shows $m = 5$ in the very first stage of MSB when all inputs are at their low levels l_j with $j = 1, ..., k = 26$, whereas $m = 18$ in the last stage when estimating the individual effect β_{17}; see Kleijnen (2015, p. 167). Such small m values do not give powerful tests of normality. However, the users may obtain additional replications. For example, we decide to increase m from 5 to 100 in the first MSB stage. There are several goodness-of-fit tests; see Kleijnen (2015, p. 91). However, the most powerful statistic for testing normality is the *Shapiro-Wilk* statistic (say) W, according to Razali and Wah (2011). This W ranges between zero and one; a small W leads to rejection of the

normality hypothesis. To compute W and its p-value, we use the free-of-charge Matlab function `swtest` documented in (BenSaïda, 2014). Our case study has $n = 2$ outputs, denoted by CT and NT. We find $W^{(\mathrm{CT})} = 0.9755$ with $p^{(\mathrm{CT})} = 0.568$ and $W^{(\mathrm{NT})} = 0.9924$ with $p^{(\mathrm{NT})} = 0.8496$, so we do not reject the normality hypothesis.

Each combination of simulation inputs gives a specific distribution (with its own parameter values) for a given simulation output type. It would be too expensive to simulate at least 100 replications for each combination. Therefore it is practical to assume that if one input combination gives normal outputs, then all other combinations of the given simulation model do so too.

In practice, the users may find it too expensive to obtain many replications of any input combination. Then these users may decide to simply assume that the simulation outputs are normally distributed—provided these outputs are the averages of long simulation runs. Such runs generate autocorrelated time series, but the output may still be normally distributed if a so-called *functional central limit theorem* holds; see Kleijnen (2015, p. 90). However, if the output is an estimated extreme quantile (e.g., the 99% quantile), then this theorem may not apply.

3. Specific MSB assumptions: three tests

The first test—called test 1—considers only the inputs that MSB found to be *important*. This test has already been detailed in Shi et al. (2014a), but we clarify some of its elements and compare test 1 with two other tests—called test 2 and test 3. The latter two tests focus on the remaining *unimportant* inputs, and are not mentioned in Shi et al. (2014a).

When MSB stops, it has identified (say) k_{I} *important inputs* and k_{U} *unimportant inputs* where $k_{\mathrm{I}} + k_{\mathrm{U}} = k$ (MSB assumes $k_{\mathrm{I}} \ll k$; see Section 1). MSB denotes an input as "important" if this input has a significant first-order effect for at least one of the n output types. An input is "unimportant" if this input does not have a significant first-order effect for any of the n output types. The heredity assumption implies that these unimportant inputs have no second-order effects for any output type. MSB finishes with estimates of the individual first-order effects of the important inputs. MSB does not estimate the second-order effects of the—important and unimportant—inputs (the MSB uses foldover designs so these second-order effects do not bias the estimated first-order effects; see Kleijnen 2015 and Shi et al. 2014).

3.1. Test 1: important inputs

Test 1 is inspired by Bettonvil and Kleijnen (1997). First, test 1 estimates the *individual* effects in the second-degree polynomial for the k_{I} inputs that MSB found to be important. We denote the

number of effects in this polynomial by $q(k_I)$. Obviously, (1) implies that $q(k_I)$ has the value $1 + k_I + k_I(k_I - 1)/2 + k_I$. If sparsity applies, then k_I is so small that estimation of these $q(k_I)$ effects does not require a screening design. Test 1 uses the most popular classic design for second-order polynomials; namely, a CCD (details on CCDs are given in (Kleijnen, 2015, pp. 64-66)). We denote its number of input combinations by n_{CCD} (the DOE literature usually denotes the number of input combinations by n; the symbol n_{CCD}—with a subscript—should not be confused with the symbol n that we use for the number of output types; see (1)). Furthermore, we must select the number of replications (say) m_{CCD}; we shall discuss this selection below.

To run the simulation with these k_I important inputs, we also need values for all the k_U *unimportant inputs*. To reduce the variances of the estimated $q(k_I)$ effects of the important inputs, we keep all k_U unimportant inputs *constant*, and we use *common random numbers* (CRN)—as follows. We keep the unimportant inputs at their coded value 0 (the "central" standardized value) if these inputs are quantitative. If unimportant inputs are qualitative, then—rather arbitrarily—we fix their levels at +1. We use CRN in the CCD; i.e., replication r (with $r = 1, ..., m_{CCD}$) of all n_{CCD} combinations uses the same pseudo-random numbers (PRNs). Altogether, we obtain estimated second-order polynomials that model the effects of the k_I important inputs on the n output types (n outputs are jointly generated by each simulation run with a specific combination of the k inputs).

Next we test the *null-hypothesis* ($H_0^{(y)}$) stating that these estimated n polynomials $\widehat{y}^{(l)}$ implied by (1) are adequate predictors of the simulation outputs (say) w:

$$H_0^{(y)} : E(\widehat{y}^{(l)}) = E(w^{(l)}) \text{ versus } H_1^{(y)} : E(\widehat{y}^{(l)}) \neq E(w^{(l)}) \tag{2}$$

where we use a superscript (y) because we shall formulate more hypotheses below. If we reject $H_0^{(y)}$, then we conclude that one or more MSB assumptions do not hold so the MSB results are debatable.

Note: Obviously, we may formulate $H_0^{(y)}$ in (2) as $E(\widehat{y}^{(l)}) - E(w^{(l)}) = 0$. A related $H_0^{(y)}$ is $|E(\widehat{y}^{(l)}) - E(w^{(l)})| \leq \Delta$ with $\Delta \geq 0$; we shall discuss the choice of Δ when discussing (4). A "pessimistic" $H_0^{(y)}$ is $|E(\widehat{y}^{(l)}) - E(w^{(l)})| \geq \Delta$. In general, the choice of H_0 is not determined by statistical reasoning. A specific example is the legal H_0: the accused is not guilty, unless proven differently. More specifically, we test $H_0^{(y)}$ *after* MSB terminates; consequently, we have good reasons to formulate the "optimistic" $H_0^{(y)}$ in (2) and later on in (9). In future research we may investigate the use of a pessimistic $H_0^{(y)}$.

To test $H_0^{(y)}$ in (2), we select (say) $n_{val} \geq 1$ combinations of the k inputs, which are declared to be either important or unimportant at the end of MSB. The selection of a specific value for n_{val}

depends on the computer time required per replication and the available computer budget. We select $n_{\text{val}} \gg 1$ combinations such that these combinations are space-filling for the quantitative inputs (either important or unimportant). We use the most popular space-filling design; namely, *Latin hypercube sampling* (LHS). Actually, LHS implies that each individual input has n_{val} different values if the input is quantitative; see Kleijnen (2015, p. 198-203). We randomly combine the n_{val} combinations of the quantitative inputs with the n_{val} values of the qualitative inputs.

We simulate these n_{val} input combinations, using m_{val} replications. To select a value for this m_{val}, the users may examine the number of replications m that MSB needed to test the significance of individual inputs in the final stage of MSB. In the case study detailed in Shi et al. (2014a) (Figure 5) m varies between 5 and 22; we decide to select $m_{\text{val}} = 10$. When simulating the n_{val} input combinations, we again use CRN.

Altogether, we use a second-degree polynomial with $q(k_{\text{I}})$ estimated parameters $\widehat{\boldsymbol{\beta}}^{(l)}$ for the k_{I} important inputs, while the remaining k_{U} unimportant inputs have zero first-order and second-order effects. Actually, MSB concluded that the k_{U} unimportant inputs have first-order effects smaller than the threshold Δ, and MSB assumes that the second-order effects of these unimportant inputs are negligible (because of heredity). Using this polynomial, test 1 predicts the output of type l for the n_{val} input combinations, and compares the average metamodel predictions

$$\overline{\widehat{y}}_i^{(l)} = \frac{\sum_{r=1}^{m_{\text{CCD}}} \widehat{y}_{i;r}^{(l)}}{m_{\text{CCD}}} \; (i = 1, ..., n_{\text{val}}) \tag{3}$$

with the corresponding average simulated output values

$$\overline{w}_i^{(l)} = \frac{\sum_{r=1}^{m_{\text{val}}} w_{i;r}^{(l)}}{m_{\text{val}}};$$

in the latter equation and in some following equations we suppress "$(i = 1, ..., n_{\text{val}})$" to simplify the notation. Because MSB should declare input j to be important if $\beta_j^{(l)} \geq \Delta_1^{(l)}$ where $\Delta_1^{(l)}$ is the *threshold of importance*, we might accept the metamodel predictor as valid if

$$|\overline{w}_i^{(l)} - \overline{\widehat{y}}_i^{(l)}| \leq \Delta_1^{(l)}. \tag{4}$$

Note: If we replaced $\Delta_1^{(l)}$ in (4) by $\Delta_0^{(l)}$ where $\Delta_0^{(l)}$ denotes the threshold of unimportance so $\Delta_0^{(l)} < \Delta_1^{(l)}$, then a test with lower type-II error probability (higher power) and higher type-I error probability would result.

However, (4) is *scale dependent*. The t-statistic defined in (8) below is scale-independent because it accounts for the estimated standard deviations (standard errors) $s(\overline{w}_i^{(l)})$ and $s(\overline{\widehat{y}}_i^{(l)})$, computed

as follows. We use the classic estimator

$$s^2(\overline{w}_i^{(l)}) = \frac{\sum_{r=1}^{m_{\text{val}}}(w_{i;r}^{(l)} - \overline{w}_i^{(l)})^2}{(m_{\text{val}} - 1)m_{\text{val}}}. \tag{5}$$

The metamodel predictors are $\widehat{y}_{i;r}^{(l)} = \mathbf{x}_i'\widehat{\boldsymbol{\beta}}_r^{(l)}$ where $\mathbf{x}_i$ denotes the vector with the values of the independent variables determined by the CCD for the important inputs, and $\widehat{\boldsymbol{\beta}}_r^{(l)}$ denotes $\widehat{\boldsymbol{\beta}}^{(l)}$ computed from replication r ($r = 1, ..., m_{\text{CCD}}$). The following variance estimator allows unequal output variances and CRN:

$$s^2(\overline{\widehat{y}}_i^{(l)}) = \frac{\sum_{r=1}^{m_{\text{CCD}}}(\widehat{y}_{i;r}^{(l)} - \overline{\widehat{y}}_i^{(l)})^2}{(m_{\text{CCD}} - 1)m_{\text{CCD}}}. \tag{6}$$

Test 1 uses the following Student t-statistic with v degrees of freedom, for input combination i and output type l:

$$t_{i;v}^{(l)} = \frac{\max\left(|\overline{w}_i^{(l)} - \overline{\widehat{y}}_i^{(l)}| - \Delta_1^{(l)},\ 0\right)}{\sqrt{s^2(\overline{w}_i^{(l)}) + s^2(\overline{\widehat{y}}_i^{(l)})}} \tag{7}$$

where $v = \min(m_{\text{val}} - 1,\ m_{\text{CCD}} - 1)$; selecting such a v is also proposed in Kleijnen (2015, pp. 115) and below (3.7) in Kleijnen (1992). Because i in (7) runs from 1 through n_{val}, this equation gives n_{val} observations on t_v per output type l (there are n output types). Therefore we use Bonferroni's inequality; i.e., we replace the classic α value by $\alpha/(n_{\text{val}} \times n)$ and reject $H_0^{(y)}$ if

$$\max_{i;l} t_{i;v}^{(l)} > t_{\nu;1-\alpha/(n_{\text{val}} \times n)} \ (i = 1, \ldots, n_{\text{val}}; l = 1, \ldots, n). \tag{8}$$

If we do not reject the estimated second-order polynomial with its k_{I} inputs, then we do not reject Assumption 1 formulated in (1).

Moreover we test *Assumption 2*, as follows. The estimated polynomials include estimates of the individual k_{I} first-order effects, so it is easy to test their assumed nonnegative *signs*. Consider

$$H_0^{(\beta)} : \beta_j^{(l)} \geq 0 \text{ versus } H_1^{(\beta)} : \beta_j^{(l)} < 0 \text{ with } j = 1, \ldots, k_{\text{I}}. \tag{9}$$

To test $H_0^{(\beta)}$, we compute the analogues of (3) and (6):

$$\overline{\widehat{\beta}}_j^{(l)} = \frac{\sum_{r=1}^{m_{\text{CCD}}}\widehat{\beta}_{j;r}^{(l)}}{m_{\text{CCD}}} \text{ and } s^2(\overline{\widehat{\beta}}_j^{(l)}) = \frac{\sum_{r=1}^{m_{\text{CCD}}}(\widehat{\beta}_{j;r}^{(l)} - \overline{\widehat{\beta}}_j^{(l)})^2}{m_{\text{CCD}}(m_{\text{CCD}} - 1)}.$$

Next, we compute

$$t_{m_{\text{CCD}}-1}^{(l)} = \frac{\overline{\widehat{\beta}}_j^{(l)}}{s(\overline{\widehat{\beta}}_j^{(l)})}. \tag{10}$$

and—analogously to (8) and remembering that $t_{v;\alpha} < 0$ if $0 < \alpha < 0.5$—we reject $H_0^{(\beta)}$ if

$$\max_l t^{(l)}_{m_{\text{CCD}}-1} \leq t_{m_{\text{CCD}}-1;\alpha/n} \ (l = 1, \ldots, n). \tag{11}$$

If we do not reject the estimated second-order polynomials with only the k_{I} important inputs, then we do not reject Assumption 3 on *heredity*; i.e., the k_{U} unimportant inputs have indeed no important first-order and second-order effects. (Remember that LHS in the n_{val} combinations varies both the important and the unimportant inputs.)

Note: If we do not reject $H_0^{(y)}$ and $H_0^{(\beta)}$, then we may use all $n_{\text{CCD}} + n_{\text{val}}$ input combinations—with their m_{CCD} and m_{val} replications—to *re-estimate* the second-order polynomial with the k_{I} important inputs. Next, we may use this re-estimated polynomial to estimate the optimal combination of the k_{I} important inputs while keeping the k_{U} quantitative unimportant inputs at their central values; e.g., response surface testology (RSM) uses second-order polynomials. Obviously, we have then entered the *post-screening* phase of the simulation experiment.

3.2. Test 2: unimportant inputs

MSB finds k_{U} *unimportant* inputs that have "nearly" zero effects in the n second-order polynomials. More precisely, MSB gives $\beta_j^{(l)} \leq \Delta_0^{(l)}$ with $j = 1$, ..., k_{U} and $\Delta_0^{(l)}$ the threshold of unimportance for output type l. Because of heredity (Assumption 3), MSB assumes that these k_{U} inputs have no important second-order effects. So in test 2 we do not estimate the $q(k_{\text{U}})$ *individual* effects of the k_{U} unimportant inputs; now we test whether these $q(k_{\text{U}})$ effects are virtually zero. In test 2 we keep all the k_{I} important inputs fixed—e.g., we fix these inputs at their base levels—so these inputsbecome constants. To simplify our explanation, we assume that all k_{I} important inputs are *quantitative* and are fixed at their coded values 0. Furthermore, we now simulate only *extreme* combinations of the k_{U} unimportant inputs. We relabel the k inputs such that MSB declares the first k_{U} inputs among the k inputs to be unimportant. We detail this test for simulation models with a single ($n = 1$) output type so MSB reduces to SB. First we explain test 2 when testing first-order effects; next we explain test 2 when testing second-order effects. Finally, we explain test 2 for MSB with two output types ($n = 2$, as in the Chinese case-study).

3.2.1 Testing first-order effects of unimportant inputs

In SB with a single output type, test 2 requires us to simulate only the two *extreme* combinations of the k_{U} unimportant inputs in which all k_{U} unimportant inputs are fixed at (a) their *low* levels so $\mathbf{x}_{\text{U}} = -\mathbf{1}$, and (b) their *high* levels so $\mathbf{x}_{\text{U}} = \mathbf{1}$, respectively. For combination (a) the metamodel

in (1) gives

$$E(y \mid \mathbf{x}_{\mathrm{U}} = -\mathbf{1}) = \beta_0 - \sum_{j=1}^{k_{\mathrm{U}}} \beta_j + \sum_{j=1}^{k_{\mathrm{U}}} \sum_{j'=j}^{k_{\mathrm{U}}} \beta_{j;j'}, \tag{12}$$

and for combination (b) this metamodel gives

$$E(y \mid \mathbf{x}_{\mathrm{U}} = \mathbf{1}) = \beta_0 + \sum_{j=1}^{k_{\mathrm{U}}} \beta_j + \sum_{j=1}^{k_{\mathrm{U}}} \sum_{j'=j}^{k_{\mathrm{U}}} \beta_{j;j'}. \tag{13}$$

Together, the latter two equations give

$$\frac{E(y \mid \mathbf{x}_{\mathrm{U}} = \mathbf{1}) - E(y \mid \mathbf{x}_{\mathrm{U}} = -\mathbf{1})}{2} = \sum_{j=1}^{k_{\mathrm{U}}} \beta_j = \beta_{1-k_{\mathrm{U}}} \tag{14}$$

where we use the notation in Shi et al. (2014a); namely, $\beta_{j-j'} = \sum_{g=j}^{j'} \beta_g$ (the left-hand side uses the endash $-$, not to be confused with the minus sign $-$) to denotes the sum of the first-order effects of the inputs j through j'.

We denote the number of replications for these two combinations by m_{val}; this m_{val} may have the same value as m_{val} in (5). We use CRN, to reduce the noise in the estimator of

$$\delta = \frac{E(w|\mathbf{x}_{\mathrm{U}} = \mathbf{1}) - E(w|\mathbf{x}_{\mathrm{U}} = -\mathbf{1})}{2}. \tag{15}$$

So we compute the difference between the simulation outputs of the two extreme combinations in replication r with $r = 1, ..., m_{\mathrm{val}}$:

$$d_r = \frac{w_r(\mathbf{x}_{\mathrm{U}} = \mathbf{1}) - w_r(\mathbf{x}_{\mathrm{U}} = -\mathbf{1})}{2}. \tag{16}$$

Because of the CRN, we compute the *t-statistic for paired differences*

$$t_{m_{\mathrm{val}}-1} = \frac{\overline{d} - E(d)}{s(d)/\sqrt{m_{\mathrm{val}}}} \tag{17}$$

with the classic estimators of the mean and variance of d

$$\overline{d} = \frac{\sum_{r=1}^{m_{\mathrm{val}}} d_r}{m_{\mathrm{val}}} \text{ and } s^2(d) = \frac{\sum_{r=1}^{m_{\mathrm{val}}} (d_r - \overline{d})^2}{m_{\mathrm{val}} - 1}. \tag{18}$$

This t-statistic gives a confidence interval (CI) for the mean difference δ defined in (15). We use this CI to test

$$H_0^{(d)} : E(d) \leq \Delta \text{ versus } H_1^{(d)} : E(d) > \Delta \tag{19}$$

where $\leq$ implies a one-sided hypothesis—because the first-order effects are not negative (Assumption 2)—and we use

$$\Delta = k_{\mathrm{U}} \Delta_0 \tag{20}$$

where Δ_0 is the SB threshold for unimportant inputs. So we expect that SB declares an individual input to be unimportant if its effect is Δ_0; together, the k_U unimportant inputs might have a total effect of $k_U\Delta_0$. Altogether, we accept bigger differences between the outputs for the two extreme input combinations, as the number of unimportant inputs increases; also see (14). So in (17) we replace $E(d)$ using (19) and (20), so we obtain

$$t_{m_{val}-1} = \frac{\overline{d} - k_U\Delta_0}{s(d)/\sqrt{m_{val}}}. \tag{21}$$

If this $t_{m_{val}-1}$ is higher than $t_{m_{val}-1;1-\alpha}$, then we reject $H_0^{(d)}$ so we reject the adequacy of the second-order polynomial with important inputs only.

Note: Because test 2 uses only two (extreme) combinations, it cannot estimate the individual effects of the k inputs (collected in β) so it cannot test $H_0^{(\beta)}$ defined in (9) (test 1 uses $n_{CCD} \geq q(k_I)$ combinations to estimate effects).

3.2.2 Testing second-order effects of unimportant inputs

We also test whether the *heredity* assumption holds, which implies that the k_U unimportant inputs have no second-order effects $\beta_{j;j'}$ with $j = 1, ..., k_U$ and $j' = j, ..., k_U$. Unfortunately, our test of the two extreme combinations (a) and (b) using (21) is completely insensitive to these $\beta_{j;j'}$; i.e., even if $\beta_{j;j'} \neq 0$, then these $\beta_{j;j'}$ do not affect the result in (14). We therefore simulate—besides the two extreme combinations—the *center combination* $\mathbf{x}_0 = \mathbf{0}$ where $\mathbf{0}$ denotes the k_U-dimensional vector with all elements equal to zero. If the heredity assumption does not hold, then the second-order polynomial defined in (1) with $n = 1$ implies $E(y \mid \mathbf{x}_U = \mathbf{0}) \neq E(y \mid \mathbf{x}_U = -\mathbf{1}) = E(y \mid \mathbf{x}_U = \mathbf{1})$. For simplicity we assume that the number of replications for $\mathbf{x}_0 = \mathbf{0}$ equals m_{val} for the two extreme combinations. We again use the CRN that are also used for the two extreme combinations. This gives the following difference between the the average of the mean simulation outputs at the two extreme combinations and the mean at the center:

$$\delta_0 = \frac{E(w \mid \mathbf{x}_U = \mathbf{1}) + E(w \mid \mathbf{x}_U = -\mathbf{1})}{2} - E(w \mid \mathbf{x}_U = \mathbf{0}). \tag{22}$$

If the second-order polynomial for the k_U unimportant inputs holds, then—*whatever* the magnitudes and signs of their first-order effects are—(22) becomes

$$\delta_0 = \sum_{j=1}^{k_U} \sum_{j'=j}^{k_U} \beta_{j;j'}. \tag{23}$$

The total number of second-order effects $\beta_{j;j'}$ (interactions and purely quadratic effects) in (23) is $k_U(k_U - 1)/2 + k_U$. Some $\beta_{j;j'}$ may be negative and some may be positive, so we do not make any

assumptions about the magnitude of the sum in (23). To estimate this δ_0, we compute

$$d_{0;r} = \frac{w_r(\mathbf{x}_U = -\mathbf{1}) + w_r(\mathbf{x}_U = \mathbf{1})}{2} - w_r(\mathbf{x}_U = \mathbf{0}) \text{ with } r = 1, ..., m_{val}. \quad (24)$$

This gives the analogue of (17):

$$t_{0;m_{val}-1} = \frac{\bar{d}_0 - E(d_0)}{s(d_0)/\sqrt{m_{val}}}. \quad (25)$$

Because individual second-order effects may be negative or positive, we define

$$H_0^{(d_0)} : E(d_0) = 0 \text{ versus } H_1^{(d_0)} : E(d_0) \neq 0 \quad (26)$$

To test $H_0^{(d_0)}$, we combine (25) and (26) and obtain the following statistic:

$$t_{0;m_{val}-1} = \frac{\bar{d}_0}{s(d_0)/\sqrt{m_{val}}}. \quad (27)$$

We use a two-sided test; i.e., if $|t_{0;m_{val}-1}|$ exceeds $t_{m_{val}-1;1-\alpha/2}$, then we reject $H_0^{(d_0)}$ so we reject the adequacy of the second-order polynomial with important inputs only.

Note: $H_0^{(d)}$ defined in (19) uses $\Delta = k_U\Delta_0$, whereas $H_0^{(d_0)}$ in (26) uses 0. If we rejct $H_0^{(d_0)}$, then we conclude that the inputs that MSB declared to be unimportant actually have important second-order effects so heredity (Assumption 3) does not hold.

When there are $n \geq 2$ output types, then we deal with each output type successively. Therefore, test 2 requires only $2n$ (extreme) combinations plus the center combination, whereas test 1 requires $n_{CCD} + n_{val}$ combinations; this n_{CCD} is determined through the (rather inefficient) CCD for the k_I important inputs, and n_{val} is selected to make the design (possibly determined through LHS) space filling so this n_{val} will be rather arbitrary and high.

3.3. Test 3: input groups and unimportant inputs

Like test 2, test 3 focuses on the unimportant inputs—but test 3 takes advantage of the existence of *input groups*; by definition, changing inputs in a group from -1 to +1 increases all n outputs or decreases all n outputs (for a thorough explanation of input groups we refer to Shi and Kleijnen (2015)). Using such input groups may save simulation effort, because MSB then estimates the effects of each input group for all n output types *simultaneously*. The readers may skip test 3 (and proceed to Section 3.4) if they are not interested in the MSB variant with input groups.

3.3.1 Testing first-order effects of unimportant inputs in input grouping

Let q denote the number of inputs groups that MSB distinguishes. The formation of the original q input groups may change when MSB finishes and declares inputs to be either important or

unimportant. Let q_{U} denote the number of input groups formed *only* by the k_{U} unimportant inputs. Let $\beta^{(1)}_{(k_{p-1}+1)-k_p}$ denote the sum of the first-order effects for output type 1 (which is the type in which the users are most interested) of input group p (p = 1, ..., q_{U}); i.e., input group p contains inputs k_{p-1} + 1, k_{p-1} + 2, ..., k_p (so the individual input k_{p-1} is the last individual input of input group $p-1$). Shi et al. (2014a) proves two theorems (called Theorems 1 and 2) that enable the estimation of this $\beta^{(1)}_{(k_{p-1}+1)-k_p}$ for all output types *simultaneously*, using replication r. In (29) and (30) displayed below, the superscript $1 \to l$ means that output type l (with l = 2, 3, ..., n) is observed "for free" when observing output type 1; i.e., running an input combination to observe output type 1 also generates an observation on output type l, so (28) and (29) or (30) have completely corresponding terms:

$$\widehat{\beta}^{(1)}_{k_{p-1}+1-k_p;r} = \frac{[w^{(1)}_{k_p;r} - w^{(1)}_{-k_p;r}] - [w^{(1)}_{k_{p-1};r} - w^{(1)}_{-(k_{p-1});r}]}{4}, \tag{28}$$

and either

$$\widehat{\beta}^{(l)}_{k_{p-1}+1-k_p;r} = \frac{[w^{(1\to l)}_{k_p;r} - w^{(1\to l)}_{-k_p;r}] - [w^{(1\to l)}_{k_{p-1};r} - w^{(1\to l)}_{-(k_{p-1});r}]}{4} \ (l = 2, 3, \ldots, n), \tag{29}$$

or

$$\widehat{\beta}^{(l)}_{k_{p-1}+1-k_p;r} = -\frac{[w^{(1\to l)}_{k_p;r} - w^{(1\to l)}_{-k_p;r}] - [w^{(1\to l)}_{k_{p-1};r} - w^{(1\to l)}_{-(k_{p-1});r}]}{4} \ (l = 2, 3, \ldots, n); \tag{30}$$

so if—within input group p—output types 1 and l have identical signs (either + or −), then $\beta^{(1)}_{(k_{p-1}+1)-k_p}$ and $\beta^{(l)}_{(k_{p-1}+1)-k_p}$ are estimated by (28) and (29); if they have opposite signs + and −, then their estimators are obtained by (28) and (30). Next we compute

$$\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}};r} = \sum_{p=1}^{q_{\mathrm{U}}} \widehat{\beta}^{(l)}_{k_{p-1}+1-k_p;r}. \tag{31}$$

Because (28) has four terms in the numerator, it might seem that MSB needs to simulate four input combinations for the estimation of a single input group p (p = 1, ..., q_{U}); i.e., it seems that MSB needs $4q_{\mathrm{U}}$ input combinations to compute $\widehat{\beta}^{(l)}_{1-k_{\mathrm{U}};r}$ in (31). However, MSB uses some input combinations applied for one input group also for another input group; namely, the two adjacent input groups, which share a boundary and use two common input combinations. In general, Shi and Kleijnen (2015, pp. 3-8) proves two more theorems (called Theorems 3 and 4) stating that MSB needs only $2q$ with $q \le n$ input combinations to estimate each individual input group effect $\beta^{(l)}_{(k_{p-1}+1)-k_p}$ (p = 1, ..., q; l = 1, ..., n) and their sum $\beta^{(l)}_{1-k}$.

3.3.2 Testing second-order effects of unimportant inputs in input grouping

Like we do in test 2, we now tests the heredity assumption through an analogue of (24). However, unlike test 2, we do not simulate the two extreme input combinations $\mathbf{x}_{\mathrm{U}}^{(l)} = -\mathbf{1}$ and $\mathbf{x}_{\mathrm{U}}^{(l)} = \mathbf{1}$ *for each output type* l, but we compute $d_{0;r}^{(l)}$ through *input groups* so we may save simulation effort.

To estimate $\beta_{1-k_{\mathrm{U}}}^{(l)}$ in (31), MSB simulates the two input combinations on the boundaries between two input groups. Now we use these combinations to obtain $d_0^{(l)}$. It is easy to show that the k inputs form q groups determined by q boundaries. However, there is a *special* boundary for a specific output type that partitions the k inputs into two opposite groups; namely, the inputs in the group above this boundary have plus signs only, and the remaining inputs in the group below the boundary have minus signs only. If the special boundary of output type l is p, then an estimator based on replication r with $r = 1, ..., m_{\mathrm{val}}$ is

$$d_{0;r}^{(1\to l)} = \frac{w_{k_p;r}^{(1\to l)} + w_{-k_p;r}^{(1\to l)}}{2} - w_r(\mathbf{x}_{\mathrm{U}} = \mathbf{0}) \text{ with } l = 2, 3, \ldots, n, \tag{32}$$

where $w_{k_p;r}^{(1\to l)}$ and $w_{-k_p;r}^{(1\to l)}$ are the two observations at boundary p on output type l when the inputs 1 through k_p are at output 1's high level and the remaining inputs are at output 1's low level, and when the input 1 through k_p are at output 1's low level and the remaining inputs are at output 1's high level.

Note: Outputs 1 and l have the same plus sign above the boundary p, whereas they have opposite signs below the boundary. So, $w_{k_p;r}^{(1\to l)}$ and $w_{-k_p;r}^{(1\to l)}$ are actually the two extreme combinations (all inputs are high, and all inputs are low, respectively) for output l and thereby are identical to $w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = \mathbf{1})$ and $w_r(\mathbf{x}_{\mathrm{U}}^{(l)} = -\mathbf{1})$ in (24).

Note: Altogether, tests 2 and 3 give the same test, using the same estimators for testing the first-order and second-order effects of unimportant inputs, but test 3 requires only $2q$ (instead of $2n$) input combinations besides the center combination. Nevertheless, we do not prefer test 3 always: (i) the cost of sorting inputs to form input groups may be relatively high, especially when n is small; (ii) the users may find test 3 more difficult to understand and implement (so before Section 3.3.1 we wrote that the readers may skip test 3).

3.4. Monte Carlo experiment with three tests

In this section, we design and analyze a Monte Carlo (MC) experiment that quantifies the performance of the three tests described in the preceding section. We base this experiment on the experiment in Shi et al. (2014a); however, the latter experiment evaluates only test 1, whereas we also evaluate tests 2 and 3.

Table 1: Combinations of MC factors, and the resulting number of replications

	MC factors			Number of simulationobservations		
Combi	n	q_U	Inputs (1-2, 3-80, 81-90, 91-98, 99-100)	Test 1	Test 2	Test 3
1	2	1	$(\mathbf{5,0,0,0,5;10,0,0,0,10})$	350	40	20
2	2	1	$(\mathbf{5,1,1,1,5;10,2,2,2,10})$	350	40	20
3	2	1	$(\mathbf{5,2,2,2,5;10,4,4,4,10})$	350	40	20
4	2	1	$(\mathbf{5,3,3,3,5;10,6,6,6,10})$	350	40	20
5	2	2	$(\mathbf{5,0,0,0,5;10,0,0,-0,-10})$	350	40	40
6	2	2	$(\mathbf{5,1,1,1,5;10,2,2,-2,-10})$	350	40	40
7	2	2	$(\mathbf{5,2,2,2,5;10,4,4,-4,-10})$	350	40	40
8	2	2	$(\mathbf{5,3,3,3,5;10,6,6,-6,-10})$	350	40	40
9	3	1	$(\mathbf{5,0,0,0,5;10,0,0,0,10;15,0,0,0,15})$	350	60	20
10	3	1	$(\mathbf{5,1,1,1,5;10,2,2,2,10;15,3,3,3,15})$	350	60	20
11	3	1	$(\mathbf{5,2,2,2,5;10,4,4,4,10;15,6,6,6,15})$	350	60	20
12	3	1	$(\mathbf{5,3,3,3,5;10,6,6,6,10;15,9,9,9,15})$	350	60	20
13	3	2	$(\mathbf{5,0,0,0,5;10,0,0,0,10;15,0,0,-0,-15})$	350	60	40
14	3	2	$(\mathbf{5,1,1,1,5;10,2,2,2,10;15,3,3,-3,-10})$	350	60	40
15	3	2	$(\mathbf{5,2,2,2,5;10,4,4,4,10;15,6,6,-6,-15})$	350	60	40
16	3	2	$(\mathbf{5,3,3,3,5;10,6,6,6,10;15,9,9,-9,-15})$	350	60	40
17	3	3	$(\mathbf{5,0,0,0,5;10,0,0,-0,-10;15,0,-0,-0,-15})$	350	60	60
18	3	3	$(\mathbf{5,1,1,1,5;10,2,2,-2,-10;15,3,-3,-3,-15})$	350	60	60
19	3	3	$(\mathbf{5,2,2,2,5;10,4,4,-4,-10;15,6,-6,-6,-15})$	350	60	60
20	3	3	$(\mathbf{5,3,3,3,5;10,6,6,-6,-10;15,9,-9,-9,-15})$	350	60	60

Note. Symbol "–" before a number means negative effect on output l.

3.4.1 Designing the experiment

We use the second-order polynomial defined in (1), fixing k (number of simulation inputs) at 100, excluding CRN, fixing α (prespecified nominal type-I error rate) at 0.05, and obtaing 1,000 macroreplications (by definition macroreplications use different PRNs, while fixing all other input values; e.g., α is fixed at 0.05 in all macroreplications). Furthermore, we control the sizes of β (polynomial coefficients or input effects) and σ_w (constant variance of simulation output w). We select only one of the combinations in Shi et al. (2014a) (namely, combination 3). This combination implies that the noise $e^{(l)}$ is normally distributed with mean 0 and standard deviation $\sigma_w = 5$. Among the $k = 100$ first-order effects there are only 4 "important" effects that correpond with the simulation inputs 1, 2, 99, and 100. For test 1 we seelct m_{CCD}, n_{val}, and m_{val} all equal to 10. Unlike Shi et al. (2014a), we investigate three MC factors (starting from our combination 3); namely, n (number of output types), q_U (number of groups with unimportant inputs), and sizes and signs of the first-order effects $\beta_j^{(l)}$. For more details we refer to the lefthand side (excluding the last three colums) of Table 1 and Shi and Kleijnen (2015). Next we present the efficiency of the three tests in Section 3.4.2, and their effectiveness in Section 3.4.3.

3.4.2 Efficiency of tests 1, 2, and 3

To quantify the efficiency of the three tests, we use the total number of simulation observations required by the three tests; see the last three columns of Table 1. Obviously, test 1 is the least efficient, in all 20 combinations. Our explanation is that test 1 requires fitting a second-order polynomial for the $k_{\rm I}$ important simulation inputs, which uses a CCD; e.g., combination 1 of the MC factors requires 350 observations where 350 is the sum of $n_{\rm CCD} \times m_{\rm CCD}$= $25 \times 10 = 250$ and $n_{\rm val} \times m_{\rm val} = 10 \times 10 = 100$ where 250 is the number needed to fit a second-order polynomial to the $k_{\rm I} = 5$ important inputs found by MSB. Furthermore, the number of simulation observations in test 2 is never smaller than the number in test 3; e.g., combinations 1 through 4 show $q_{\rm U} = 1 < n = 2$ so the number of observations in test 2 is $2n \times m_{\rm val} = 4 \times 10 = 40$ and the number in test 3 is $2q_{\rm U} \times m_{\rm val} = 2 \times 10 = 20$. In general, test 3 is more efficient than test 2 if $q_{\rm U} < n$; tests 2 and 3 are equally efficient if $q_{\rm U} = n$. In practice, the users know whether $q_{\rm U} = n$ or $q_{\rm U} < n$, so they do know which test is more efficient.

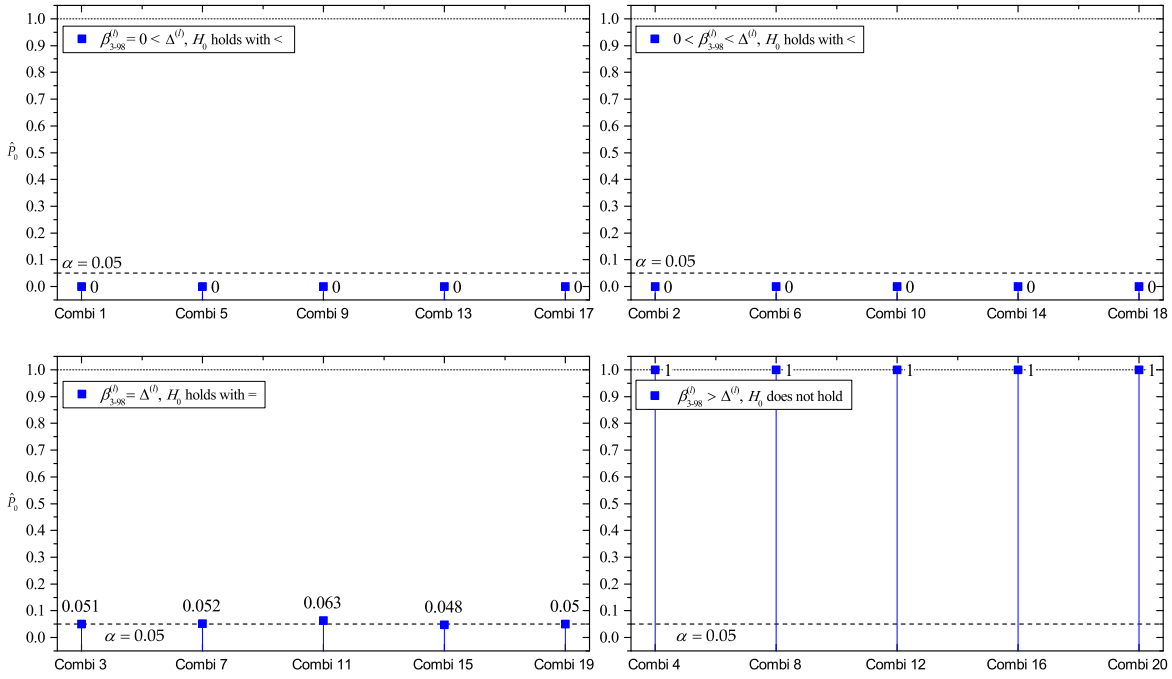
3.4.3 Effectiveness of tests 1, 2, and 3

To quantify the effectiveness of the three tests, we estimate the probability of rejecting an important effect of an input group or a single input. Let p_0 denotes this probability of rejecting $H_0^{(\beta)}$ defined in (9) for a single input, and defined analogously for $\beta_{j-j'}^{(l)}$. Using 1,000 macroreplications, we record the percentage of macroreplications that rejects $H_0^{(\beta)}$. If $H_0^{(\beta)}$ holds, then a test should ideally give $\widehat{p}_0 = \alpha$ (where α is the nominal type-I error probability). First we compute $\widehat{p}_0$ for various magnitudes of the first-order effects of the inputs declared to be "unimportant"; next we compute $\widehat{p}_0$ for various magnitudes of the second-order effects of these "unimportant" inputs. Obviously, $1 - \widehat{p}_0$ estimates the Type-II error rate.

Figure 1 presents $\widehat{p}_0$ in the following combinations listed in Table 1: {1, 5, 9, 13, 17}, {2, 6, 10, 14, 18}, {3, 7, 11, 15, 19}, and {4, 8, 12, 16, 20}. We present these $\widehat{p}_0$ only for test 2 because test 3 gives similar results. The x-axis lists specific combinations and the y-axis gives the corresponding $\widehat{p}_0$; e.g., the upper-left plot gives $\widehat{p}_0$ for the combination {1, 5, 9, 13, 17}, which have exactly zero aggregated effects for the $k_{\rm U} = 96$ unimportant inputs so these aggregated effects are much smaller than $\Delta^{(l)} = 96\Delta_0^{(l)}$. This figure shows that $\beta_{1-k_U}^{(l)}$ strongly influences $\widehat{p}_0$; e.g., the upper two plots show $\widehat{p}_0 = 0$ if $\beta_{3-98}^{(l)} < \Delta^{(l)}$, the lower-right plot shows $\widehat{p}_0 = 1$ if $\beta_{3-98}^{(l)} > \Delta^{(l)}$ and the lower-left plot shows $\widehat{p}_0 \approx \alpha = 0.05$ if $\beta_{3-98}^{(l)} = \Delta^{(l)}$ (so $\beta_{3-98}^{(l)}$ reaches its maximum while $H_0^{(\beta)}$still holds).

From these plots we conclude that tests 2 and 3 give appropriate type-I and type-II error rates. We do not display results for test 1, because this test turns out to give relatively high $\widehat{p}_0$ when

Figure 1: $\widehat{p}_0$ of Method 2 for various combinations of first-order effects



there are considerably many unimportant inputs (high k_{U}). Our explanation is that test 1 uses (7), which has the term $\Delta_1^{(l)}$ so it does not consider the *aggregated* effects of the unimportant simulation inputs. These aggregated effects may increase the difference between $\overline{w}_i^{(l)}$ and $\overline{\widehat{y}}_i^{(l)}$; this difference increases p_0, as more unimportant inputs are involved (higher k_{U}).

Whereas Table 1 implies that all second-order effects of the unimportant inputs are exactly zero, we now allow *non-zero second-order effects* so that we can investigate the heredity assumption for tests 2 and 3. We start from combination 17 in Table 1, so all first-order effects of the unimportant inputs for the $n = 3$ output types are exactly zero. Next we investigate the following four cases for the second-order effects of these unimportant inputs.

Case 1: The heredity assumption does hold, so input j has the first-order effect $\beta_j^{(l)} = 0$ and second-order effects $\beta_{j;j'}^{(l)} = 0$ with $j \leq j' = 1, ..., k_{\mathrm{U}}$. Actually, Case 1 is identical to combination 17, in which the unimportant inputs labeled 3 through 98 have zero first-order and second-order effects. Consequently, the definition in (23) gives $\delta_0^{(l)} = 0$. Therefore, we expect $\widehat{p}_0 \leq \alpha$ for Case 1.

Case 2: The unimportant input #10 has the first-order effect $\beta_{10}^{(l)} = 0$, but its purely quadratic effect is $\beta_{10;10}^{(l)} = c\Delta_1^{(l)}$ where c equals one of the following six values: 0.01, 0.1, 1, 25, 50, 100 ($\Delta_1^{(l)}$ still denotes the "importance" threshold). We expect that if c increases, then $\widehat{p}_0$ exceeds α more and more—until $\widehat{p}_0$ reaches its maximum value of 1.

Case 3: The unimportant inputs #10 and #20 have purely quadratic effects that may be much

higher than their first-order effects—but these quadratic effects have opposite signs; i.e., $\beta^{(l)}_{10;10} = c\Delta^{(l)}_1$ and $\beta^{(l)}_{20;20} = -c\Delta^{(l)}_1$. Consequently, these quadratic effects cancel out so $\delta^{(l)}_0 = 0$. Therefore, we expect $\widehat{p}_0 = \alpha$ even if c is high; i.e., our test has little power in this case.

Case 4: Each unimportant input has a zero first-order effect (so $\beta^{(l)}_j = 0$ with $j = 1, ..., k_U$) and has relatively small second-order effects so $\beta^{(l)}_{j;j'} = c\Delta^{(l)}_0$ where c has (very small) values from 0.0001 to 0.05 ($\Delta^{(l)}_0$ is the "unimportance" threshold). The aggregated effects of the unimportant inputs may still be high, if there are many unimportant inputs (so k_U is high), which makes $\delta^{(l)}_0$ high—see (23). Therefore, we expect $\widehat{p}_0$ to vary between the desired value α and the maximum value 1, as c or k_U increases.

For each of these four cases the 1,000 macroreplications give $\widehat{p}_0$, which now denotes the percentage of macroreplications that rejects $H^{(d_0)}_0$ in (26). Because tests 2 and 3 give almost the same $\widehat{p}_0$, we display $\widehat{p}_0$ only for test 2. Case 1 gives $\widehat{p}_0 = 0.049$, which is very close to the desired value $\alpha = 0.05$—as we expected. Cases 2 through 4 give the six estimated power curves in Figure 2 for various c; the first two plots (in the first row) display $\widehat{p}_0$ for Cases 2 and 3, while the remaining four plots give $\widehat{p}_0$ for Case 4 with $k_U = 10, 20, 40, 80$. This figure demonstrates that $\delta^{(l)}_0$ has an important effect on $\widehat{p}_0$, as we detail in the next four comments.

(i) If $\delta^{(l)}_0 = 0$, then $\widehat{p}_0 \approx \alpha = 0.05$; e.g., in Case 3 (upper-right plot), the two quadratic effects cancel out so $\delta^{(l)}_0 = 0$ and all observed values for $\widehat{p}_0$ (see the squares) are close to the dashed line that corresponds with $\alpha = 0.05$—no matter how much c changes.

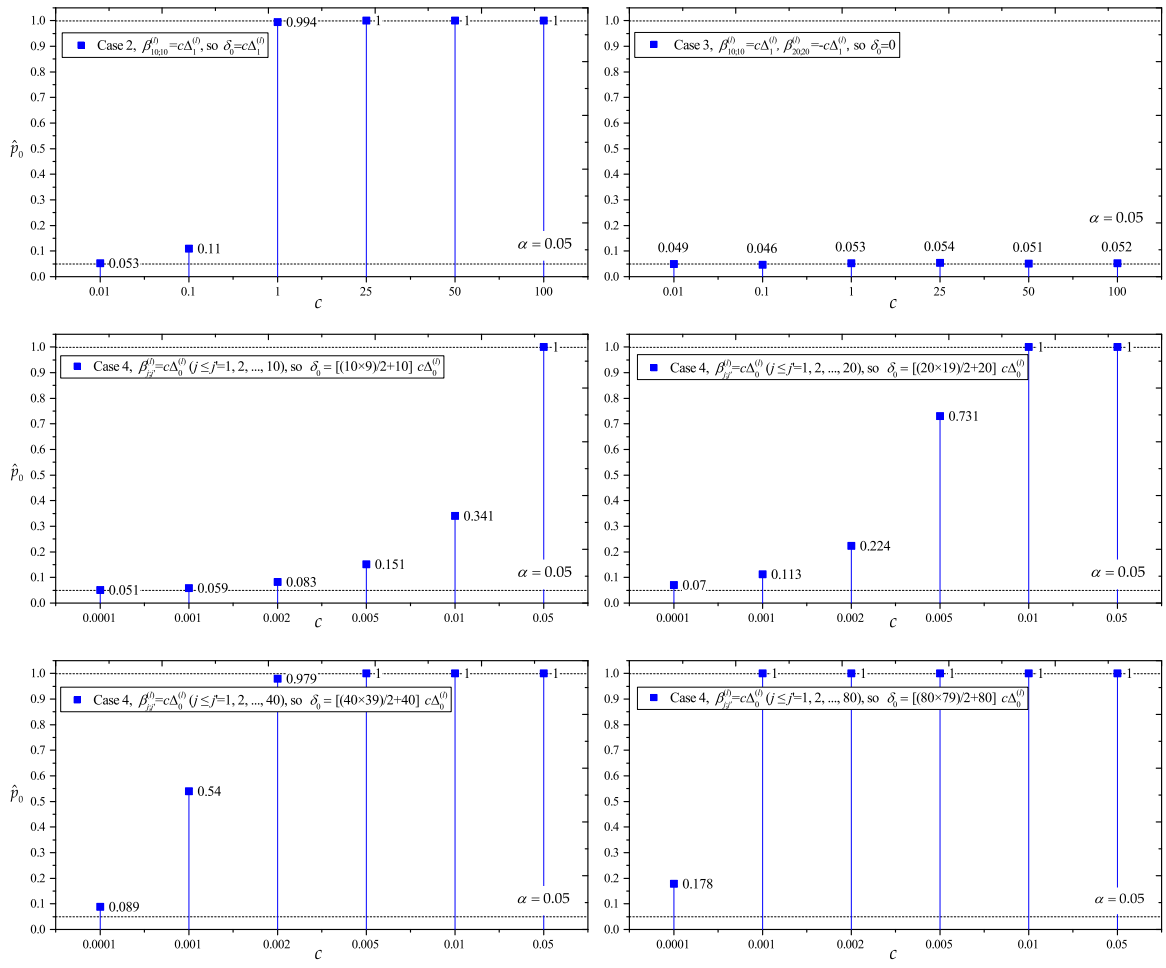
(ii) If $\delta^{(l)}_0 \geq \Delta^{(l)}_1$, then $\widehat{p}_0 \uparrow 1$; e.g., in Case 2 (upper-left plot), c= 1 so $\delta^{(l)}_0 = \beta^{(l)}_{10;10} = \Delta^{(l)}_1$ gives $\widehat{p}$ as high as 0.994. Moreover, in Case 4 (each "unimportant" input has second-order effects) even a small k_U or c gives $\widehat{p}_0 = 1$ because the sum $\delta^{(l)}_0$ may still exceed $\Delta^{(l)}_1$; e.g., $k_U = 10$ and $c = 0.05$ (the rightmost point in the middle-left plot) gives $\delta^{(l)}_0 = [(10 \times 9)/2 + 10] \times 0.05 \times \Delta^{(l)}_0 = 2.75\Delta^{(l)}_0$ $> \Delta^{(l)}_1$, so $\widehat{p}_0 = 1$.

(iii) If $0 < \delta^{(l)}_0 \leq \Delta^{(l)}_1$, then $\alpha < \widehat{p}_0 \leq 1$; e.g., Case 4 with $k_U = 40$ and $c = 0.001$ (the second point in the lower-left plot) gives $\delta^{(l)}_0 = [(40 \times 39)/2 + 40] \times 0.001 \times \Delta^{(l)}_0 = 0.82\Delta^{(l)}_0 < \Delta^{(l)}_1$, and $\widehat{p}_0 = 0.341 < 1$. Similarly, $\widehat{p}_0 = 0.731 < 1$ if $k_U = 20$ and $c = 0.005$ (the fourth point in the middle-right plot) so $\delta^{(l)}_0 = 1.05\Delta^{(l)}_0 < \Delta^{(l)}_1$.

(iv) The more $\delta^{(l)}_0$ approximates $\Delta^{(l)}_1$, the bigger $\widehat{p}_0$ becomes; e.g., $\delta^{(l)}_0 = 0.82\Delta^{(l)}_0 < \delta^{(l)}_0 = 1.05\Delta^{(l)}_0$ gives $\widehat{p}_0 = 0.341 < \widehat{p}_0 = 0.731$. Similar results are found for other $\widehat{p}_0$-values between 0.05 and 1.

We conclude that tests 2 and 3 have so much power that they detect most cases that violate the heredity assumption—except for Case 3, in which two quadratic effects cancel out exactly, but

Figure 2: Estimated power $\widehat{p}_0$ of Method 2 for various cases with second-order effects



we find Case 3 rather pathological, so we do not further discuss this case.

4. Case study: a logistics simulation in China

Whereas the MC experiment in the preceding section enabled us to estimate the performance of the proposed three tests in a controlled laboratory setting, we now present a case study to investigate the performance in practice. This case study was originally detailed in Shi et al. (2014b), and may be summarized as follows.

The given discrete-event simulation model represents a Chinese *third-party logistics* (TPL) company that wants to improve the *just-in-time* (JIT) system for its customer; this customer is a car manufacturer. The TPL company expects to open another assembly plant. When this new plant will open, the current TPL capacity will not meet the logistic needs. Management wants to main-

tain the current logistic performance, measured through the *average cycle time* (CT) of a part and the *number of throughput* (NT) per "month" or 30-day period. A high CT conflicts with the JIT philosophy. NT is the sum of the shipments collected at the part suppliers and delivered to the assembly plants per month. The goal of this case study is to identify the inputs that are important for one or both performance measures (CT, NT).

The simulation has $k = 26$ inputs. Actually, $k = 26$ is not a high value in screening, but a second-degree polynomial with $k = 26$ implies $q = 378$ effects so the CCD should have at least 378 combinations (which also need to be replicated); i.e., MSB provides a practical screening design. Inputs 1 through 5 are known to have plus signs for both outputs, whereas the remaining 21 inputs have opposite signs for the two outputs; namely, plus for CT and minus for NT. Consequently, we can define two *input groups*; namely, group 1 with inputs 1 though 5, and group 2 with the remaining inputs (labeled 6 through 26). Shi et al. (2014a) uses Wan et al. (2010)'s SPRT with $\Delta_1^{(CT)} = 5$ and $\Delta_0^{(CT)} = 2.5$, and $\Delta_1^{(NT)} = 3,000$ and $\Delta_0^{(NT)} = 2,000$. Both SB and MSB find five important inputs; namely, the inputs labelled 4, 5, 14, 17, and 20. So, inputs 4 and 5 are in input group 1, and inputs 14, 17, and 20 are in input group 2. Furthermore, SB and MSB declare the same inputs to be important; i.e., SB identifies the inputs 4, 5, 14, 17, and 20 for CT and input 17 for NT. SB requires 355 simulation observations, whereas MSB requires only 233 observations.

4.1. Test 1: case-study results

Shi et al. (2014a) applies test 1, using a CCD for the $k_I = 5$ inputs that SB and MSB declare to be important. This CCD enables the estimation of the 21 (= 1 + 5 + 10 + 5) individual effects in this polynomial. The number of replications per CCD combination is $m_{CCD} = 10$, which is based on the number of replications in the last stages of SB and MSB displayed in Shi et al. (2014a, Fig. 5). The unimportant quantitative inputs are fixed at their coded value 0, and the one unimportant qualitative input (a priority rule) is fixed at +1, which denotes first-in-first-out or FIFO (the default queueing rule of the current supply-chain). CRN are used by replication r ($r = 1, ..., m_{CCD}$) of all the CCD combinations.

The resulting polynomials give $R^2 = 0.9608$ and $R^2_{adj} = 0.9519$ for CT, and $R^2 = 0.9641$, and $R^2_{adj} = 0.9588$ for NT, whereas first-order polynomials give $R^2 = 0.7022$ and $R^2_{adj} = 0.6683$ for CT, and $R^2 = 0.6988$ and $R^2_{adj} = 0.6733$ for NT. So, the estimated second-order polynomials are much better, and seem *adequate* for predicting the simulation outputs. Actuasllly, we might compute other validation statistics besides R^2 and R^2_{adj}; e.g., cross-validation statistics (see Kleijnen (2015, p. 112–121).

Given that these polynomials are adequate, it makes sense to examine their individual estimated coefficients $\widehat{\beta}$. It turns out that the *signs* of the estimated first-order effects of the important inputs agree with the assumed signs; namely, inputs 4 and 5 have minus signs for both CT and NT, and inputs 14, 17, and 20 have opposite signs for these outputs. So test 1 does not reject the assumption of known signs for all first-order effects of the important inputs.

Table 2: Test 1 in new combination i

i	1	2	3	4	5	6	7	8	9	10
$\overline{w}_i^{CT}$	28.09	31.27	25.06	45.22	42.02	67.57	24.30	38.63	27.25	58.65
$s^2(\overline{w}_i^{CT})$	1.38	4	0.38	0.80	0.02	1.70	0.26	1.92	1.04	1.30
$\overline{\widehat{y}}_i^{CT}$	34.03	30.01	26.53	49.40	40.78	61.26	26.26	34.37	24.96	54.58
$s^2(\overline{\widehat{y}}_i^{CT})$	0.28	0.57	0.77	1.22	0.50	1.76	0.64	0.72	0.41	0.38
$t_{i;9}^{CT}$	2.28	0	0	0.83	0	1.78	0	0.78	0	0
$\overline{w}_i^{NT}$	49,387	53,664	53,122	45,513	51,563	38,952	51,003	44,424	48,402	51,562
$s^2(\overline{w}_i^{NT})$	64,407	209,913	36,151	9,250	52,819	23,850	97,016	30,896	21,505	3,876
$\overline{\widehat{y}}_i^{NT}$	46,738	52,531	51,475	43,665	50,991	40,323	51,397	45,007	51,669	48,317
$s^2(\overline{\widehat{y}}_i^{NT})$	52,303	35,195	32,936	43,594	38,718	26,054	49,384	35,292	52,609	22,877
$t_{i;9}^{NT}$	0	0	0	0	0	0	0	0	0.98	1.50

To test that all first-order and second-order effects of all *unimportant* inputs are zero, Shi et al. (2014a) selects $n_{\text{val}} = 10$ new combinations (after the n_{CCD} old combinations). These new combinations are selected through LHS with uniform sampling of values between -1 and 1 for all 25 quantitative unimportant inputs, and sampling the two values -1 and 1 for the one qualitative unimportant input. For each combination, Shi et al. (2014a) selects the number of replications to be $m_{\text{val}} = 20$. Altogether, these 10 LHS combinations with their 20 replications give the simulated $\overline{w}$ and the predicted $\overline{\widehat{y}}$ and their estimated variances $s^2(\overline{w})$ and $s^2(\overline{\widehat{y}})$ displayed in Table 2. These statistics give $t_{i;v}^{(l)}$, which denote the Studentized prediction errors defined in (7) with $v = \min(10-1, 20-1) = 9$. Furthermore, Shi et al. (2014a) selects $\alpha = 0.20$; such a relatively high α value is typical when applying Bonferroni's inequality. So, the critical value becomes $t_{10-1;(0.20/(10\times 2))} = t_{9;0.01} = 2.821$. The table shows that $\max_{i;l} t_i^{(l)} = 2.28$ (this 2.28 is found in column $i = 1$ for CT), but this maximum value is not significant. We conclude that in this case study test 1 does not reject the three assumptions of SB and MSB.

4.2. Tests 2 and 3: case-study results

As we mentioned above, Shi et al. (2014a) finds $k_{\text{U}} = 21$ *unimportant* inputs. These inputs imply $q_{\text{U}} = 2$ *input groups*; namely, input group 1 comprising inputs 1 through 3, and input group 2 comprising the remaining 18 unimportant inputs. Because q_{U} equals n (number of output types), tests 2 and 3 require the same number of extreme input combinations—namely, $2n = 2q = 4$—to

Table 3: Validation results of Tests 2 and 3

	$d^{(\mathrm{CT})}$	$d^{(\mathrm{NT})}$	$d_0^{(\mathrm{CT})}$	$d_0^{(\mathrm{NT})}$
$\overline{d}$	17.05	10,223.73	10.43	-4,643.33
$s(d)$	1.92	922.18	2.92	915.24

test the first-order effects. To test the second-effects (featuring in the heredity assumption), tests 2 and 3 need one more combination; namely, the center point. To enable a fair comparison with test 1, we use tests 2 and 3 with the same $m_{\mathrm{val}} = 20$. Altogether, the number of simulation observations is $(4 + 1)\times 20 = 100$. We use a "per comparison" error rate $\alpha = 0.05$ (test 1 used a "familywise" rate of $\alpha = 0.20$; different types of error rates are discussed in Kleijnen (2015, p. 98).

The definitions of $d_r^{(\mathrm{CT})}$ and $d_r^{(\mathrm{NT})}$ ($r = 1, ..., m_{\mathrm{val}}$) follow from (16). These $d_r^{(\mathrm{CT})}$ and $d_r^{(\mathrm{NT})}$ give $\overline{d}$ and $s(d)$, using (18); see columns 2 and 3 of Table 3; we shall discuss $d_{0;r}^{(\mathrm{CT})}$ and $d_{0;r}^{(\mathrm{NT})}$ displayed in columns 4 and 5, below.

Using (21) with $\Delta^{(\mathrm{CT})} = k_{\mathrm{U}}\Delta_0^{(\mathrm{CT})} = 52.5$ and $\Delta^{(\mathrm{NT})} = k_{\mathrm{U}}\Delta_0^{(\mathrm{NT})} = 42,000.0$, we obtain $t_{19}^{(\mathrm{CT})} = -82.6$ and $t_{19}^{(\mathrm{NT})} = -154.1$. These negative values are much smaller than the positive critical value $t_{20-1;1-0.05/2} = t_{19;0.975} = 2.093$ where we use /2 because there are $n = 2$ outputs and we use Bonferroni's inequality. So, we do not reject H_0 in (19) for CT and NT.

Note: The preceding test neglects the possibility of a first-order effect on (say) CT of one input declared to be unimportant, that is actually higher than the threshold $\Delta_0^{(\mathrm{CT})} = 2.5$ while the other unimportant inputs have zero first-order effects. The sum of the $k_{\mathrm{U}} = 21$ unimportant inputs equals $\overline{d}^{(\mathrm{CT})} = 17.05$, so if these inputs had the same first-order effects, then these estimated effects would be $17.05/21 = 0.81$—which is considerably less than $\Delta_0^{(\mathrm{CT})} = 2.5$. Furthermore, $\overline{d}^{(\mathrm{CT})} = 17.05$ is small compared with the sum of the first-order effects of the $k = k_{\mathrm{I}} + k_{\mathrm{U}} = 26$ inputs on CT; namely, $\widehat{\beta}_{1-26}^{(\mathrm{CT})} = 46.41$. The sum of the first-order effects of the $k_{\mathrm{I}} = 5$ important inputs is 31.7. Analogous results hold for the other output, NT.

The last two columns of Table 3 display $\overline{d}_0$ and $s(d_0)$ for CT and NT; see the definition in (24). The two-sided test in (25) gives $|t_{0;19}^{(\mathrm{CT})}| = 15.93$ and $|t_{0;19}^{(\mathrm{NT})}| = 22.69$. Selecting $\alpha = 0.05$, we obtain the critical value $t_{20-1;1-0.05/(2\times 2)} = t_{19;0.9875} = 2.4334$, which is much smaller so we reject $H_0^{(d)}$ in (19).

The results of test 1 suggested that a second-order polynomial with only the important inputs adequately explains the effects of the simulation inputs on the simulation outputs; i.e., the unimportant inputs seem to have small first-order and second-order effects. Tests 2 and 3, however, suggest that there are many of these small second-order effects so their sum is statistically signifi-

cant. Altogether, tests 2 and 3 require only a few simulation observations, but may give misleading results; so, next we may apply the more expensive test 1 to test the assumptions of MSB.

5. Conclusions and future research

MSB is applied to search for the k_I important inputs among the k inputs of a given simulation model with $n \geq 1$ types of simulation responses (outputs). We suppose that k is so high that the fitting of a second-order polynomial with $q = 1 + k + k(k-1)/2 + k$ effcets requires an unacceptable large number of simulation combinations; moreover, these combinations must be replicated to estimate the noise of the simulation outputs. Factor screening assumes sparsity: k_I/k is small; e.g., 20% according to the 20-80 rule (in our case study, $k_I/k = 5/26 = 0.19$). MSB assumes that the simulation responses are normally distributed—like many other statistical methods assume—and that n second-order polynomials with k inputs are adequate approximations—or "valid metamodels" of the input/output function that is implicitly defined by the given simulation model—where the k first-order effects have known signs, and unimportant first-order effects imply unimportant second-order effects (two-factor interactions and purely quadratic effects). In this paper we discussed how we can test these assumptions, as follows.

To test normality of the simulation outputs. we may start MSB generating (say) 100 replications of one of the extreme combinations of the k inputs; e.g. the combination with all inputs at their low levels for one of the output types. Next we may apply the Shapiro-Wilk statistic, and assume that its result applies to all combinations. Instead of testing normality, we might simply assume that normality holds if these outputs are the averages of long simulation runs.

After MSB stops, we may choose among three tests for the specific MSB assumptions; namely, test 1—originally proposed in Shi et al. (2014a)—or one of two novel tests, called tests 2 and 3.

Test 1 uses a CCD for the k_I important inputs found by MSB. This CCD is much smaller than a CCD for all k inputs; e.g., our case study gives $k_I = 5$ so $q(k_I) = 21$ individual effects must be estimated through a CCD, whereas $k = 26$ implies $q(k) = 378$. Each CCD combination is replicated, applying CRN. To test whether the estimated polynomial with k_I inputs is an adequate approximation, we may compute classic statistical validation statistics; e.g., R^2_{adj}. The estimated individual effects in this polynomial enable us to test the signs of the first-order effects of the important inputs. To test that all first-order and second-order effects of all unimportant inputs are indeed zero, we select some new combinations—after the old CCD combinations—applying LHS. We again replicate each new combination, applying CRN. This enables us to compute the Studentized prediction errors.

Test 2 and *test 3* focus on the $k_U = k - k_I$ *unimportant* inputs. The basic assumption of screening implies that k_U is so high that a CCD is impracticable. Therefore these two tests simulate only very few combinations. Test 2 simulates the two extreme combinations for a given output type, plus the center combination. Test 3 is rather complicated, but may be more efficient than test 2, because the former test uses so-called input groups. Unfortunately, the k_U inputs may have "unimportant"—but not exactly zero—first-order effects, and there are so many of these effects that together they give significant differences between the simulation responses of the two extreme combinations. Comparison with the response at the center combination is hampered by the high number of interactions and purely quadratic effects; moreover, these effects may be negative or positive.

We conclude that after MSB ends, we should obtain replicated simulation responses for a CCD with the k_I important inputs only. If test 1 does not reject the three specific MSB assumptions, then we may start the *post-screening* phase; i.e., we may use all old and new combinations to re-estimate the second-order polynomials or to estimate a Kriging metamodel; both metamodel types are also used in Law (2015, pp. 668-679). Next we we may use these metamodels to estimate the optimal input combination for the simulation model.

We hope that our article will stimulate researchers to further explore the validation and follow-up of MSB. For example, further reserach is needed to select the number of replications for the CCD and the LHS for test 1. Furthermore, we may investigate *deterministic* simulation models; e.g., we may replace the Studentized predicitin error by the relative predicition error $\widehat{y}/w$. Moreover, if we reject the MSB assumptions, then we may apply a more expensive screening test, such as Morris's test; see Kleijnen (2015) and Shi et al. (2016).

Acknowledgment

This work is partly supported by the National Natural Sciences Foundation of China under Grants No 71402048, 71372134 and 71671060, and the China Postdoctoral Science Foundation Funded Project No 2015M582228.

References

BenSaïda, A., 2014. Shapiro-Wilk and Shapiro-Francia normality tests. MATLAB Central [online] 15.

Bettonvil, B. W. M., 1990. Detection of important factors by sequential bifurcation. Ph.D. thesis, Universiteit van Tilburg.

Bettonvil, B., Kleijnen, J. P. C., 1997. Searching for important factors in simulation models with many factors: Sequential bifurcation. European Journal of Operational Research 96 (1), 180–194.

Han, W., Rajan, P., Frazier, P. I., Jedynak, B. M., 2017. Bayesian group testing under sum observations: A parallelizable two-approximation for entropy loss. IEEE Transactions on Information Theory 63 (2), 915–933.

Kleijnen, J. P. C., 1992. Regression metamodels for simulation with common random numbers: comparison of validation tests and confidence intervals. Management Science 38 (8), 1164–1185.

Kleijnen, J. P. C., 2015. Design and Analysis of Simulation Experiments, 2nd Edition. Springer US, New York.

Kleijnen, J. P. C., Shi, W., 2017. Sequential probability ratio tests: conservative and robust, Working paper, Department of Management, Tilburgh University.

Law, A. M., 2015. Simulation Modeling and Analysis, 6th Edition. McGraw-Hill, Boston.

Martín, R. A. G., Sánchez, J. M. G., 2015. Screening for important factors in large-scale simulation models: some industrial experiments. Bachelor Degree Project in Automation Engineering, The University of Skövde, Skövde, Sweden.

Razali, N. M., Wah, Y. B., 2011. Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests. Journal of Statistical Modeling and Analytics 2 (1), 21–33.

Shi, W., Kleijnen, J. P. C., 2015. Validating the assumptions of sequential bifurcation in factor screening. Working paper, Department of Management, Tilburg University.

Shi, W., Kleijnen, J. P. C., Liu, Z., 2014a. Factor screening for simulation with multiple responses: Sequential bifurcation. European Journal of Operational Research 237 (1), 136 – 147.

Shi, W., Shang, J., Liu, Z.-X., Zuo, X.-L., 2014b. Optimal design of the auto parts supply chain for jit operations: Sequential bifurcation factor screening and multi-response surface methodology. European Journal of Operational Research 236, 664–676.

Shi, W., Shang, J., Zhang, Z., 2016. Simulation screening and false discovery rate control for both main and interaction effects. In: Roeder, T. M. K., Frazier, P. I., Szechtman, R., Zhou,

E., Huschka, T., Chick, S. E. (Eds.), Proceedings of the 2016 Winter Simulation Conference. Institute of Electrical and Electronics Engineers, Inc., pp. 512–521.

Wan, H., Ankenman, B. E., Nelson, B. L., 2010. Improving the efficiency and efficacy of controlled sequential bifurcation for simulation factor screening. INFORMS Journal on Computing 22 (3), 482–492.

Wu, J. C. F., Hamada, M., 2009. Experiments: Planning, Analysis, and Optimization, 2nd Edition. Wiley, New York.